WHAT IS CLAIMED IS

1. A method for building at least part of an application dynamically, comprising the steps of:

providing a document including a specification for building at least part of an application; and

building said at least part of an application dynamically using said specification; wherein said at least part of an application is new and said building includes providing said at least part of an application.

2. The method of claim 1, wherein said document is written in semi-structure data.

3. The method of claim 2, wherein said document is written in an XML based language.

4. The method of claim 1, further comprising the step of:

launching an application including said at least part.

5. The method of claim 1, wherein said at least part of an application is a class and said specification is a class descriptor.

6. The method of claim 5, further comprising the steps of:

parsing said document; and

translating said parsed document into at least one class descriptor.

7. The method of claim 1, wherein said step of providing includes the step of:

creating said at least part of an application.

8. The method of claim 7, wherein said at least part of an application is a class, said specification is a class descriptor, and said step of creating includes the step of:

generating said class.

9. The method of claim 8, wherein said step of generating includes the steps of:

creating a new class; and

filling in class details into said new class, according to said class descriptor.

10. The method of claim 9, wherein said class descriptor includes at least one method descriptor; and said step of generating further includes the steps of:

for each method descriptor included in said class descriptor:

creating a new method; and

filling in method details into said method according to said method descriptor.

11. The method of claim 10, wherein said at least one method descriptor includes at least one code line descriptor, and said step of generating further includes the steps of:

for each code line descriptor:

creating a new code line; and

filling in code line details into said code line according to said code line descriptor.

12. The method of claim 9, wherein said class descriptor includes at least one property descriptor; and said step of generating further includes the steps of:

for each property descriptor included in said class descriptor:

creating a new property; and

filling in property details into said property according to said property descriptor.

13. The method of claim 5, wherein said class contains at least one other class.

14. The method of claim 13, wherein said class descriptor includes at least one property descriptor, and said at least one property descriptor describes said at least one other class.

15. The method of claim 13, wherein at least one of said at least one other class is generated.

16. The method of claim 13, wherein at least one of said at least one other class is loaded.

17. The method of claim 16, wherein said at least one of said at least one other class that is loaded was downloaded from a remote location.

18. The method of claim 16, wherein said at least one of said at least one other class that is loaded was an existing locally available class.

19. The method of claim 5, further comprising the step of:

instantiating an object from said class.

20. The method of claim 19, wherein said step of instantiating includes the steps of:

creating a new object;

associating said new object with said class; and

creating properties of said new object according to property descriptors included in said class descriptor.

21. The method of claim 1, wherein said step of providing a document includes the step of:

receiving said document via a network..

22. The method of claim 1, wherein said step of providing said at least part of an application includes the step of:

receiving said at least part via a network.

23. The method of claim 22, wherein said step of receiving includes the steps of:

detecting that said at least part is not available locally;

and

downloading said at least part via said network.

24. The method of claim 23, wherein said at least part is a basic part.

25. A system for dynamically building at least part of an application, comprising:

a generator for extracting from a document a specification for building at least part of an application; and

a loader for dynamically building said at least part of an application based on said specification, wherein said at least part of an application is new and said building includes providing said at least part of an application.

26. The system of claim 25, further comprising:

a local registry for storing information about at least part of said at least part of an application.

27. The system of claim 26, further comprising:

a registrar for managing said registry.

28. The system of claim 25, further comprising:

a storage for providing at least part of said at least part of an application to said loader.

29. The system of claim 25, further comprising:

a repository for storing said at least part of an application when active.

30. The system of claim 25, further comprising:

a communication connection for receiving said document from a remote server.

31. The system of claim 30, wherein said communication connection is also used for receiving said at least part of an application.

32. The system of claim 25, wherein said at least part is a class, the system further comprising:

an object repository for storing an instantiated object of said class when active.

33. A method for building an application of interest, comprising the steps of:

receiving a document including a specification for at least part of an application, over a network; and

building the application of interest;

wherein said document allows the application of interest to be built without the transfer of an executable file of the application of interest over said network.

34. The method of claim 33, further comprising the step of:

receiving executables for parts of the application of interest in a set-up phase prior to said step of receiving a document.

35. The method of claim 34, wherein said executables received in said set-up phase are basic parts.

36. The method of claim 33, wherein said step of building includes the step of:

receiving executables for parts of the application of interest over said network.

37. The method of claim 36, wherein said executables are for basic new parts.

38. The method of claim 36, wherein said parts of the application of interest are classes.

39. The method of claim 33, wherein said step of building is accomplished with insignificant accompanying transfer over said network of executables for parts of the application of interest.

40. The method of claim 33, wherein the step of building includes the step of:

creating parts of the application of interest.

41. The method of claim 40, wherein said step of creating includes the step of:

generating classes.

42. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for building at least part of an application dynamically, comprising the steps of:

providing a document including a specification for building at least part of an application; and

building said at least part of an application dynamically using said specification; wherein said at least part of an application is new and said building includes providing said at least part of an application.

43. A computer program product comprising a computer useable medium having computer readable program code embodied therein for building at least part of an application dynamically , the computer program product comprising:

computer readable program code for causing the computer to provide a document including a specification for building at least part of an application; and

computer readable program code for causing the computer to build said at least part of an application dynamically using said specification; wherein said at least part of an application is new and said building includes providing said at least part of an application.

44. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for building an application of interest, comprising the steps of:

receiving a document including a specification for at least part of an application, over a network; and

building the application of interest;

wherein said document allows the application of interest to be built without the transfer of an executable file of the application of interest over said network.

45. A computer program product comprising a computer useable medium having computer readable program code embodied therein for building an application of interest, the computer program product comprising:

computer readable program code for causing the computer to receive a document including a specification for at least part of an application, over a network ; and

computer readable program code for causing the computer to build the application of interest;

wherein said document allows the application of interest to be built without the transfer of an executable file of the application of interest over said network.

46. The method of claim 1, further comprising the preliminary step of a server sending said document.

47. The method of claim 25, further comprising a server for sending said document.

48. The method of claim 33, further comprising the preliminary step of a server sending said document.